

**METHOD AND APPARATUS FOR UPDATING DATABASE OF AUTOMATIC
SPELLING CORRECTIONS**

FIELD OF THE INVENTION

This invention pertains to spell checking on a computer and more particularly to automatically updating the wordlist of spelling corrections to automatically correct in a document.

BACKGROUND OF THE INVENTION

Over the years since computers first entered popular society, computers have given users the ability to do tasks more quickly and efficiently. One very early application of computers was for editing and printing documents. Before computers, documents were typed on typewriters. Correcting such documents required using a dictionary to identify spelling errors, and documents would then have to be re-typed. Computers with word processing or editing software enabled documents to be changed and re-printed without the user having to re-type the document.

Early on, software companies realized the advantages of providing electronic dictionaries, to save users the effort of having to use manual dictionaries. But users had to manually invoke the spell checking component of the editor. Eventually, software companies added automatic spell checking to their editing programs. Automatic spell checking uses a static update list to let users know about misspelled words immediately, without having to run the spell checker.

FIG. 1 is a flowchart of automatic spell checking as it is used in the prior art, as exemplified by the AutoCorrect in Microsoft® Word and the QuickCorrect tool in Corel® WordPerfect®. (Microsoft is a registered trademark of Microsoft Corporation in the United States and/or other countries. Corel and WordPerfect are registered trademarks of Corel Corporation or Corel Corporation Limited.) At step 105, a newly entered word is parsed. At step 110, the word is checked to see if it is misspelled. If the entered word is misspelled, at step 115 the static update list is checked to see if the word is listed. If the misspelled word is in the static update list, then at step 120, the misspelled word is replaced with the correctly spelled word.

FIG. 2 shows what happens to a document when the static update list is used. Document 205 is a document in which the user has typed a misspelled word that is in the static update list. In document 205, the misspelled word is "teh." Note that the cursor is positioned directly after the word. The editing program cannot be certain that the word is completed until after the user has typed a space or some other symbol that ends the word. Once the user types a space, the editing program checks to see if the word is in the static update list. Since the word "teh" is in the static update list, the editing program automatically replaces the word "teh" with the correctly spelled word "the."

FIG. 2 also shows what happens to a document when the misspelled word is not in the static update list. In document 215, the user has just typed the word "copmany," which is misspelled. But because the word "copmany" is not in the static update list, the editing program cannot automatically correct the misspelling in document 220. In this case, the editing program informs the user that a misspelled word has been found, for example, by placing wavy line 225 underneath the misspelled word.

FIG. 3 shows a static update list as used for spell checking in the prior art. In FIG. 3, static update list 305 includes pairs of words. One entry in each pair is a misspelled word. The other entry is the correct spelling for the misspelled word in the first entry. For example, pair 310 includes misspelled word "teh" 315 and correctly spelled word "the" 320. Often, editing programs include pre-built static update lists, which include commonly misspelled words, as shown in FIG. 3.

Spell checking increases a user's productivity by allowing the user to know more quickly about mistakes in his/her document. But the user is still required to manually update the dictionary and static update lists. Where a user repeatedly makes the same typing mistakes that are not already in the static update list, the user must interrupt his/her typing to add new words to the static update list.

Accordingly, a need remains for a method and apparatus that helps a user update an automatic spell checker.

SUMMARY OF THE INVENTION

The invention provides a method and software for automatically updating a spell checking static update list by watching the user's typing and learning what words the user intended. When the user enters a word in a document, the word is scanned. If the word is misspelled, a static update list is checked to see if the misspelled word has an automatic

correction (i.e., a corresponding correct word). If the misspelled word is not in the static update list and the user corrects the misspelling, the misspelled word and the correction word are both captured as a word pair in a dynamic update list. If the user corrects the misspelling a sufficient number of times, the static update list can itself be dynamically updated to recognize the correction regularly entered by the user.

The foregoing and other features, objects, and advantages of the invention will become more readily apparent from the following detailed description, which proceeds with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a flowchart of how a static update list is used in the prior art.

FIG. 2 shows how static update lists are used in the prior art to automatically correct misspelled words.

FIG. 3 shows an update list according to the prior art.

FIG. 4 shows a computer system on which the invention can be implemented.

FIG. 5 shows a dynamic update module according to the invention interacting with other modules in software for spell checking an edited document.

FIG. 6 shows a document being edited by a user using a dynamic update module according to FIG. 5 in a preferred embodiment.

FIG. 7 shows a document being edited by a user using a dynamic update module according to FIG. 5 in a second embodiment.

FIG. 8 shows the dynamic update list being used to update the static update list according to the invention.

FIGS. 9A and 9B show data structures for the dynamic update module of FIG. 5.

FIGS. 10A and 10B show a flowchart of the method for maintaining the dynamic update list in the preferred embodiment.

FIG. 11 is a flowchart of the method for updating the static update list by the dynamic update module as shown in FIG. 8.

FIG. 12 is a flowchart of the method for a user to select between alternative corrections of a misspelled word in the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. 4 shows a computer system 405 on which a dynamic update list for spell checking can operate according to the invention. Computer system 405 conventionally includes a computer 410, a monitor 415, a keyboard 420, and a mouse 425. Optional equipment not shown in FIG. 4 can include a printer and other input/output devices. Also not shown in FIG. 4 are the internal components of computer system 405: e.g., a central processing unit, memory, file system, etc.

FIG. 5 shows a dynamic update module interacting with other modules in an edited document. In FIG. 5, document 505 is being edited by editor 510. Although in the preferred embodiment editor 510 is a word processing program, a person skilled in the art will recognize that other editors exist. Many other programs include editing modules: e.g., spreadsheets, database management systems, and schedulers include basic editors that can include spell checking capabilities. For the remainder of this document, however, it is assumed that the editor in question is part of a word processing program.

Other modules that can operate on document 505 are thesaurus 515, grammar checker 520, spell checker 525, and automated checker 530. Thesaurus 515 provides the user with synonyms of words in document 505. Grammar checker 520 checks document 505 for proper grammar, informs the user when possible grammar errors exist, and presents the user with correct grammar. Spell checker 525 checks document 505 for proper spelling, informs the user of misspelled words, and presents the user with correct spelling options. Automated checker 530 performs automatic spelling correction (and in an alternate embodiment grammar checking as well) on document 505 by comparing words entered by the user with static update list 540. When a word is matched with a word in static update list 540, automated checker replaces the misspelled word with the correction in static update list 540.

Dynamic update 535 watches document 505. When the user enters a misspelled word which automated checker 530 does not automatically correct, dynamic update 535 checks to see if the user corrects the misspelling. If the user corrects the misspelling, dynamic update 535 checks to see if the user has misspelled the word before and corrected it in the same way. If the user has made the misspelling and correction before, dynamic update 535 increments a count of the number of times the user has made the mistake and corrected it in dynamic update list 545. Otherwise, dynamic update 535 adds the misspelled word and its correction for future reference.

It may happen that the misspelled word is included in static update list 540. If the correction in static update list 540 does not satisfy the user, the user can overrule the correction with an alternate correction. Dynamic update 535 can then add the misspelled word and the alternate correction to dynamic update list 545 for later use.

5 If the user has made a mistake and corrected it a threshold number of times, dynamic update 535 moves the misspelled word and its correction from dynamic update list 545 to static update list 540. Then, when the user makes the same misspelling in the future, automated checker 530 will automatically correct the misspelling for the user. In the preferred embodiment, dynamic update 535 tracks the number of times the user has misspelled a word in a particular way and has corrected the misspelling to a particular correct spelling. However, a person skilled in the art will recognize that any measure indicating the correction is worth adding to static update list 540 can be used in place of a threshold count. For example, the measure indicating the correction is worth adding to static update list 540 can be a count of how many times the user has typed the correctly spelled word. If the user has typed the correctly spelled word enough times, the static update list can be updated based on only one misspelling by the user. Or the measure can be an examination of how frequently the correctly spelled the word has been entered relative to the number of words typed by the user. If the user enters the correctly spelled word enough times in a single paragraph (or page, document, or multiple documents), the static update list can be updated.

20 Before dynamic update 535 changes static update list 540, dynamic update 535 should check static update list 540 to see if static update list 540 has the misspelled word already in it, with an alternate correction. If static update list 540 includes a correction for the misspelled word, dynamic update 535 will not change static update list 540. In another embodiment, dynamic update 535 informs the user that two corrections are possible for the misspelled word, and lets the user select the preferred correction. Then, dynamic update 535 updates static update list 540 according to the user's preference.

25 In the preferred embodiment, dynamic update 535 is implemented in software. Thus, both the maintenance of dynamic update list 545 and the update of static update list 540 are done via software. However, a person skilled in the art will recognize that other implementations are possible, depending on the situation. For example, specially constructed hardware implementing a special purpose machine can include a hardware unit implementing the functions of dynamic update 535.

FIG. 6 shows a document being edited by a user using a dynamic update module according to FIG. 5 in a preferred embodiment. Document 605 shows the user typing the word "company." In document 605, the user has misspelled the word "company" as "copmany." In document 610, the automated checker has checked the word, determined that it is misspelled, and also determined that the misspelled "copmany" is not in the static update list. In document 615, the user has started to correct the spelling of "copmany." In document 620, the user has completed correcting the word "company." Finally, entry 625 in the dynamic update list is incremented to show that the user has misspelled and corrected "company" five times.

FIG. 7 shows a document being edited by a user using a dynamic update module according to FIG. 5 in a second embodiment. Document 705 shows the user typing the word "there," but misspelling the word as "theri." In document 710, the automated checker has checked the word, determined that it is misspelled, and automatically corrected the word as "their." In document 715, the user has changed the correction from "their" to "there." Because the misspelled word was already in the static update list of the automated checker, the dynamic update module presents the user with both the original correction alternative and the user's alternate correction. The user then has the option of keeping the original correction in the static update list, replacing the original correction with the user's alternate correction, or having no automatic correction for the misspelling "theri."

FIG. 8 shows the dynamic update list being used to update the static update list. In FIG. 8, dynamic update list 545 has pairs of words as entries where the user has corrected misspelled words. Entry 810 in dynamic update list 545 has been corrected by the user 10 times. As the user has set the threshold maximum count 815 at 10, the pair consisting of misspelled word "copmany" and the correct spelling "company" should now be added to static update list 540. (Note that the invention enables the user to set threshold maximum count 815 at her desired level: if desired, the user can set threshold maximum count 815 at a higher or lower level.) Entry 825 in static update list 540 is added automatically by the dynamic update module.

FIGs. 9A and 9B show alternative data structures for the dynamic update module of FIG. 5. In FIG. 9A the entries in the dynamic update module are organized as a list 905. In list 905, the first element in each entry is the misspelled word. The second element in each entry is the corrected spelling of the misspelled word. The third element is a count of the number of times the word has been misspelled and corrected. For example, entry 910

includes misspelled word "copmany" 915, which the user has corrected to correctly spelled word "company" 920 eight times according to counter 925.

FIG. 9B shows the entries in the dynamic update module organized as a tree. Tree 955 is a sorted tree, where elements to the left of a node come alphabetically before the node, and elements to the right of the node come alphabetically after the node. Although tree 955 is a very basic tree, a person skilled in the art will recognize that more complicated tree structures can be used: for example, B-trees and other tree structures that allow for simple insertion, deletion, and searching of the tree. Node 960 contains the same information as entry of 910 of FIG. 9A.

FIGs. 10A and 10B show a flowchart of the method for maintaining the dynamic update list 545 in the preferred embodiment. At step 1005, a word is parsed. At step 1010, the word is checked to see if it is misspelled. If the word is misspelled, then at step 1012 the static update list is checked to see if the misspelled word is listed. If the misspelled word is in the static update list, then at step 1013 the misspelled word is replaced with the correct spelling of the word from the static update list. At step 1014, the system checks to see if the user replaces the word with a correct spelling. Note that this can happen even if the misspelled word was replaced with a word from the static update list: the static update list may have replaced the misspelled word with a correction other than that desired by the user. At step 1015 the correct spelling of the misspelled word is received. At step 1020 the dynamic update list is checked to see if the misspelled word and its correct spelling are already in the dynamic update list. Note that the dynamic update list stores the correction of the misspelled word, even if the misspelled word was corrected from the static update list. If the misspelled word and its correct spelling are not in the dynamic update list, then at step 1025 the misspelled word and the correct spelling are added to the dynamic update list. Then, at step 1030 the count of the misspelled word and its correction are incremented.

FIG. 11 is a flowchart of the method for updating the static update list by the dynamic update module. At step 1105, the dynamic update module checks to see if the correction count of any entry in the dynamic update list exceeds the threshold count. If the threshold count is exceeded, then at step 1110, the misspelled word and its correction are added to the static update list, and at step 1115, removed from the dynamic update list.

FIG. 12 is a flowchart of the method for a user to select between alternative corrections of a misspelled word. At step 1205, the user is presented with the alternative correct spellings for the misspelled word. In the preferred embodiment, the user is presented

with the alternative correct spellings in a dialog box. At step 1210, the user selects the desired correct spelling for the static update list. At step 1215, the static update list is updated with the misspelled word and the user's chosen correct spelling. Finally, at step 1220, the misspelled word is removed from the dynamic update list.

5

EXTENSIONS

A person skilled in the art will recognize that there are many possible extensions of this invention. Some extensions included in this invention are as follows:

- Document formatting. Many users repeatedly format documents to their tastes. Such formatting includes choosing a new paper size, changing margins and fonts, inserting headers and footers, and the like. By observing how users format their documents, the invention can aid the user, either by modifying a template on which document formatting is based, or by automatically updating document formats after the user begins the formatting process. The invention can also maintain a list of different formats, and let the user select the desired format from the list.
- Other language support. Although the preferred embodiment discloses the use of the English language, the invention is equally applicable to foreign languages. Further, the invention can determine a frequency of the misspelled words for the user. If the user is typing a document in which the misspelling frequency substantially exceeds the user's norm, the invention can conclude that the document is being generated in another language, and can start comparing "misspelled" words in the document with other language dictionaries. This extension is especially useful where the user has used foreign language dictionaries before.
- Multilingual support. Some users can be expected to be multi-lingual, and will generate documents that can be either monolingual (entirely in one language) or multilingual (in more than one language). The invention can maintain multiple dynamic update lists for the user, including every combination of languages in which the user generates documents. This avoids corrections being made that are valid in one language but not in another language.
- Multi-word corrections. Although in the preferred embodiment the space character is treated as a word delimiter, it can also be treated as a regular character. Then, if the user makes a correction across a space, the dynamic update list can capture this correction and eventually add it to the static update list. For example, if the user types

10

15

20

25

30

“mor ethan” and then corrects the phrase to “more than,” this multi-word correction can be identified.

- Word completion. In an alternate embodiment, the dynamic update list can update the static update list with word completions. The dynamic update list takes a properly spelled word, such as “explanation,” and reduces it to a partial word “expl.” If the user completes the partial word “expl” as “explanation” a sufficient number of times, the static update list can receive the pair “expl” and “explanation,” allowing the user to short-hand his/her typing. Word completion can be very useful in functional documents, where a particular phrase must be entered multiple times.
- Pick lists. In an alternate embodiment, the static update list can store more than one possible misspelling correction. For example, the static update list can store both “their” and “there” as correct spellings for “theri.” The static update list can then correct a misspelling based on language context. In another embodiment, a pick list (a list of words) can pop up, either after the misspelling is determined or by user selection, from which the user can select the desired word.
- Markov processes. A Markov process is a mapping from one state to a list of second states, each second state having a positive probability, and the sum of the probabilities for all second states being one. Given the input state, the Markov process can “guess” at the subsequent state. In the context of the invention, a Markov process can be used to “guess” the correct spelling of a misspelled word based on the frequencies with which the user has corrected the misspelling to an alternate spelling. For example, if the user types “theri,” the user has corrected this misspelling to “their” 60% of the time, and the user has corrected this misspelling to “there” 40% of the time, the Markov process can “guess” with 60% certainty that the user intended “their.” Each time the user corrects a misspelling, the probabilities can be updated, affecting the probabilities of the Markov process. Markov processes also combine well with pick lists and word completion.
- Multi-user support. Many computers are used by only a single user. But some computers are used by more than one user. Static and dynamic update lists can be maintained for each user, providing each user with individual typing biases and correction preferences.
- Multiple dynamic update lists. In the preferred embodiment, a single dynamic update list is used for updating the static update list for a single user (assuming one user per

computer; see multi-user support above). This means that dynamic update lists are maintained across multiple documents for an individual user. But dynamic update lists can be tied to particular documents.

- Different thresholds. In the preferred embodiment, a counter is used to track how many times the user has entered a particular misspelled word and corrected it. But, as discussed above, other thresholds can be used to determine if the correction should be added to the static update list. For example, the static update list can be updated if the user types the correctly spelled word enough times, even if the user misspelled the word only once. Or the threshold can be set based on the frequency with which the user types the misspelling relative to the number of words in the document. The user can be allowed to choose her desired threshold technique, and can also be allowed to choose when the threshold is considered reached.

Having illustrated and described the principles of our invention in a preferred embodiment thereof, it should be readily apparent to those skilled in the art that the invention can be modified in arrangement and detail without departing from such principles. I claim all modifications coming within the spirit and scope of the accompanying claims.